

A stochastic coordinate descent primal-dual algorithm with dynamic stepsize for large-scale composite optimization

Meng Wen^{1,2}, Shigang Yue⁴, Yuchao Tang³, Jigen Peng^{1,2}

1. School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, P.R. China
2. Beijing Center for Mathematics and Information Interdisciplinary Sciences, Beijing, P.R. China
3. Department of Mathematics, NanChang University, Nanchang 330031, P.R. China
4. School of Computer Science, University of Lincoln, LN6 7TS, UK

Abstract In this paper we consider the problem of finding the minimizations of the sum of two convex functions and the composition of another convex function with a continuous linear operator. With the idea of coordinate descent, we design a stochastic coordinate descent primal-dual splitting algorithm with dynamic stepsize. Based on randomized Modified Krasnosel'skii-Mann iterations and the firmly nonexpansive properties of the proximity operator, we achieve the convergence of the proposed algorithms. Moreover, we give two applications of our method. (1) In the case of stochastic minibatch optimization, the algorithm can be applied to split a composite objective function into blocks, each of these blocks being processed sequentially by the computer. (2) In the case of distributed optimization, we consider a set of N networked agents endowed with private cost functions and seeking to find a consensus on the minimizer of the aggregate cost. In that case, we obtain a distributed iterative algorithm where isolated components of the network are activated in an uncoordinated fashion and passing in an asynchronous manner. Finally, we illustrate the efficiency of the method in the framework of large scale machine learning applications. Generally speaking, our method is comparable with other state-of-the-art methods in numerical performance, while it has some advantages on parameter selection in real applications.

* Corresponding author.

E-mail address: wen5495688@163.com

Keywords: distributed optimization; large-scale learning; proximity operator; dynamic stepsize

MR(2000) Subject Classification 47H09, 90C25,

1 Introduction

The purpose of this paper is to designing and discussing an efficient algorithmic framework with dynamic stepsize for minimizing the following problem

$$\min_{x \in \mathcal{X}} f(x) + g(x) + (h \circ D)(x), \quad (1.1)$$

where \mathcal{X} and \mathcal{Y} are two Euclidean spaces, $f, g \in \Gamma_0(\mathcal{X})$, $h \in \Gamma_0(\mathcal{Y})$, and f is differentiable on \mathcal{Y} with a β -Lipschitz continuous gradient for some $\beta \in (0, +\infty)$ and $D : \mathcal{X} \rightarrow \mathcal{Y}$ a linear transform. This parameter β is related to the convergence conditions of algorithms presented in the following section. Here and in what follows, for a real Hilbert space \mathcal{H} , $\Gamma_0(\mathcal{H})$ denotes the collection of all proper lower semi-continuous convex functions from \mathcal{H} to $(-\infty, +\infty]$. Despite its simplicity, when $g = 0$ many problems in image processing can be formulated in the form of (1.1).

In this paper, the contributions of us are the following aspects:

(I) we provide a more general iteration in which the coefficient τ, σ is made iteration-dependent to solve the general Problem (1.1), errors are allowed in the evaluation of the operators $prox_{\sigma h^*}$ and $prox_{\tau g}$. The errors allow for some tolerance in the numerical implementation of the algorithm, while the flexibility introduced by the iteration-dependent parameters τ_k and σ_k can be used to improve its convergence pattern. We refer to our algorithm as ADMMDS⁺, and when $\tau_k \equiv \tau, \sigma_k \equiv \sigma$, the ADMM⁺ algorithm introduced by Bianchi [2] is a special case of our algorithm.

(II) Based on the results of Bianchi [2], we introduce the idea of stochastic coordinate descent on modified krasnoselskii mann iterations. The form of Modified Krasnosel'skii-Mann iterations can be translated into fixed point iterations of a given operator having a contraction-like property. Interestingly, ADMMDS⁺ is a special instances of Modified Krasnosel'skii-Mann iterations. By the view of stochastic coordinate descent, we know that at each iteration, the algorithm is only to update a random subset of coordinates. Although this leads to a perturbed version of the initial Modified Krasnosel'skii-Mann

iterations, but it can be proved to preserve the convergence properties of the initial unperturbed version. Moreover, stochastic coordinate descent has been used in the literature [18-20] for proximal gradient algorithms. We believe that its application to the broader class of Modified Krasnosel'skii-Mann algorithms can potentially lead to various algorithms well suited to large-scale optimization problems.

(III) We use our views to large-scale optimization problems which arises in signal processing and machine learning contexts. We prove that the general idea of stochastic coordinate descent gives a unified framework allowing to derive stochastic algorithms with dynamic stepsize of different kinds. Furthermore, we give two application examples. Firstly, we propose a new stochastic approximation algorithm with dynamic stepsize by applying stochastic coordinate descent on the top of ADMMDS⁺. The algorithm is called as stochastic minibatch primal-dual splitting algorithm with dynamic stepsize (SMPDSDS). Secondly, we introduce a random asynchronous distributed optimization methods with dynamic stepsize that we call as distributed asynchronous primal-dual splitting algorithm with dynamic stepsize (DAPDSDS). The algorithm can be used to efficiently solve an optimization problem over a network of communicating agents. The algorithms are asynchronous in the sense that some components of the network are allowed to wake up at random and perform local updates, while the rest of the network stands still. No coordinator or global clock is needed. The frequency of activation of the various network components is likely to vary.

The rest of this paper is organized as follows. In the next section, we introduce some notations used throughout in the paper. In section 3, we devote to introduce PDSDS and ADMMDS⁺ algorithm, and the relation between them, we also show how the ADMMDS⁺ includes ADMM⁺ and the Forward-Backward algorithm as special cases. In section 4, we provide our main result on the convergence of Modified Krasnosel'skii-Mann algorithms with randomized coordinate descent. In section 5, we propose a stochastic approximation algorithm from the ADMMDS⁺. In section 6, we address the problem of asynchronous distributed optimization. In the final section, we show the numerical performance and efficiency of propose algorithm through some examples in the context of large-scale l_1 -regularized logistic regression.

2 Preliminaries

Throughout the paper, we denote by $\langle \cdot, \cdot \rangle$ the inner product on \mathcal{X} and by $\|\cdot\|$ the norm on \mathcal{X} .

Assumption 2.1. *The infimum of Problem (1.1) is attained. Moreover, the following qualification condition holds*

$$0 \in \text{ri}(\text{dom } h - D \text{ dom } g).$$

The dual problem corresponding to the primal Problem (1.1) is written

$$\min_{y \in \mathcal{Y}} (f + g)^*(-D^*y) + h^*(y),$$

where a^* denotes the Legendre-Fenchel transform of a function a and where D^* is the adjoint of D . With the Assumption 2.1, the classical Fenchel-Rockafellar duality theory [3], [10] shows that

$$\min_{x \in \mathcal{X}} f(x) + g(x) + (h \circ D)(x) = \min_{y \in \mathcal{Y}} (f + g)^*(-D^*y) + h^*(y). \quad (2.1)$$

Definition 2.1. Let f be a real-valued convex function on \mathcal{X} , the operator prox_f is defined by

$$\begin{aligned} \text{prox}_f : \mathcal{X} &\rightarrow \mathcal{X} \\ x &\mapsto \arg \min_{y \in \mathcal{X}} f(y) + \frac{1}{2} \|x - y\|_2^2, \end{aligned}$$

called the proximity operator of f .

Definition 2.2. Let A be a closed convex set of \mathcal{X} . Then the indicator function of A is defined as

$$\iota_A(x) = \begin{cases} 0, & \text{if } x \in A, \\ \infty, & \text{otherwise.} \end{cases}$$

It can easy see the proximity operator of the indicator function in a closed convex subset A can be reduced a projection operator onto this closed convex set A . That is,

$$\text{prox}_{\iota_A} = \text{proj}_A$$

where proj is the projection operator of A .

Definition 2.3. (Nonexpansive operators and firmly nonexpansive operators [3]). Let \mathcal{H} be a Euclidean space (we refer to [3] for an extension to Hilbert spaces). An operator $T : \mathcal{H} \rightarrow \mathcal{H}$ is nonexpansive if and only if it satisfies

$$\|Tx - Ty\|_2 \leq \|x - y\|_2 \text{ for all } (x, y) \in \mathcal{H}^2.$$

T is firmly nonexpansive if and only if it satisfies one of the following equivalent conditions:

- (i) $\|Tx - Ty\|_2^2 \leq \langle Tx - Ty, x - y \rangle$ for all $(x, y) \in \mathcal{H}^2$;
- (ii) $\|Tx - Ty\|_2^2 = \|x - y\|_2^2 - \|(I - T)x - (I - T)y\|_2^2$ for all $(x, y) \in \mathcal{H}^2$.

It is easy to show from the above definitions that a firmly nonexpansive operator T is nonexpansive.

Definition 2.4. A mapping $T : \mathcal{H} \rightarrow \mathcal{H}$ is said to be an averaged mapping, iff it can be written as the average of the identity I and a nonexpansive mapping; that is,

$$T = (1 - \alpha)I + \alpha S, \tag{2.2}$$

where α is a number in $]0, 1[$ and $S : \mathcal{H} \rightarrow \mathcal{H}$ is nonexpansive. More precisely, when (2.2) or the following inequality (2.3) holds, we say that T is α -averaged.

$$\|Tx - Ty\|^2 \leq \|x - y\|^2 - \frac{(1 - \alpha)}{\alpha} \|(I - T)x - (I - T)y\|^2, \forall x, y \in \mathcal{H}. \tag{2.3}$$

A 1-averaged operator is said non-expansive. A $\frac{1}{2}$ -averaged operator is said firmly non-expansive.

We refer the readers to [3] for more details. Let $M : \mathcal{H} \rightarrow \mathcal{H}$ be a set-valued operator. We denote by $\text{ran}(M) := \{v \in \mathcal{H} : \exists u \in \mathcal{H}, v \in Mu\}$ the range of M , by $\text{gra}(M) := \{(u, v) \in \mathcal{H}^2 : v \in Mu\}$ its graph, and by M^{-1} its inverse; that is, the set-valued operator with graph $(v, u) \in \mathcal{H}^2 : v \in Mu$. We define $\text{zer}(M) := \{u \in \mathcal{H} : 0 \in Mu\}$. M is said to be monotone iff $\forall (u, u') \in \mathcal{H}^2, \forall (v, v') \in Mu \times Mu', \langle u - u', v - v' \rangle \geq 0$ and maximally monotone iff there exists no monotone operator M' such that $\text{gra}(M) \subset \text{gra}(M') \neq \text{gra}(M)$.

The resolvent $(I + M)^{-1}$ of a maximally monotone operator $M : \mathcal{H} \rightarrow \mathcal{H}$ is defined and single-valued on \mathcal{H} and firmly nonexpansive. The subdifferential ∂J of $J \in \Gamma_0(\mathcal{H})$ is maximally monotone and $(I + \partial J)^{-1} = \text{prox}_J$.

Lemma 2.1. (*Krasnosel'skii-Mann iterations [3]*) Assume that $T : \mathcal{H} \rightarrow \mathcal{H}$ is $\frac{1}{\delta}$ -averaged and that the set $\text{Fix}(T)$ of fixed points of T is non-empty. Consider a sequence $(\rho_k)_{k \in \mathbb{N}}$ such that $0 \leq \rho_k \leq \delta$ and $\sum_k \rho_k(\delta - \rho_k) = \infty$. For any $x^0 \in \mathcal{H}$, the sequence $(x^k)_{k \in \mathbb{N}}$ recursively defined on \mathcal{H} by $x^{k+1} = x^k + \rho_k(Tx^k - x^k)$ converges to some point in $\text{Fix}(T)$.

Lemma 2.2. (*Baillon-Haddad Theorem [3, Corollary 18.16]*). Let $J : \mathcal{H} \rightarrow \mathbb{R}$ be convex, differentiable on \mathcal{H} and such that $\pi \nabla J$ is nonexpansive, for some $\pi \in]0, +\infty[$. Then ∇J is π -cocoercive; that is, $\pi \nabla J$ is firmly nonexpansive.

Lemma 2.3. (*Composition of averaged operators [4, Theorem 3]*). Let $\alpha_1 \in]0, 1[$, $\alpha_2 \in]0, 1[$, $T_1 \in \mathcal{A}(\mathcal{H}, \alpha_1)$, and $T_2 \in \mathcal{A}(\mathcal{H}, \alpha_2)$. Then $T_1 \circ T_2 \in \mathcal{A}(\mathcal{H}, \alpha')$, where

$$\alpha' := \frac{\alpha_1 + \alpha_2 - 2\alpha_1\alpha_2}{1 - \alpha_1\alpha_2}.$$

Proposition 2.1. ([5,6]). Let \tilde{H} be a Hilbert space, and the operators $T : \tilde{H} \rightarrow \tilde{H}$ be given. If the mappings $\{T_i\}_{i=1}^N$ are averaged and have a common fixed point, then

$$\bigcap_{i=1}^N \text{Fix}(T_i) = \text{Fix}(T_1 \cdots T_N).$$

Here the notation $\text{Fix}(T) \equiv \text{Fix}T$ denotes the set of fixed points of the mapping T ; that is, $\text{Fix}T := \{x \in \tilde{H} : Tx = x\}$.

Averaged mappings are useful in the convergence analysis, due to the following result.

Proposition 2.2. ([7]). Let $T : \tilde{H} \rightarrow \tilde{H}$ an averaged mapping. Assume that T has a bounded orbit, i.e., $\{T^k x^0\}_{k=0}^\infty$ is bounded for some $x^0 \in \tilde{H}$. Then we have:

- (i) T is asymptotically regular, that is, $\lim_{k \rightarrow \infty} \|T^{k+1}x - T^k x\| = 0$, for all $x \in \tilde{H}$;
- (ii) for any $x \in \tilde{H}$, the sequence $\{T^k x\}_{k=0}^\infty$ converges to a fixed point of T .

The so-called demiclosedness principle for nonexpansive mappings will often be used.

Lemma 2.4. (*Demiclosedness Principle [7]*). Let C be a closed and convex subset of a Hilbert space \tilde{H} and let $T : C \rightarrow C$ be a nonexpansive mapping with $\text{Fix}T \neq \emptyset$. If $\{x^k\}_{k=1}^\infty$ is a sequence in C weakly converging to x and if $\{(I - T)x^k\}_{k=1}^\infty$ converges strongly to y , then $(I - T)x = y$. In particular, if $y = 0$, then $x \in \text{Fix}T$.

Lemma 2.5. (*The Resolvent Identity [8,9]*). For $\lambda > 0$ and $\nu > 0$ and $x \in \tilde{E}$, where \tilde{E} is a Banach space,

$$J_\lambda x = J_\nu \left(\frac{\nu}{\lambda} + \left(1 - \frac{\nu}{\lambda}\right) J_\lambda x \right).$$

3 A primal-dual splitting algorithm with dynamic stepsize

3.1 Derivation of the algorithm

For Problem (1.1), Condat [1] considered a primal-dual splitting method as follows:

$$\begin{cases} \tilde{y}^{k+1} = \text{prox}_{\sigma h^*}(y^k + \sigma D x^k), \\ \tilde{x}^{k+1} = \text{prox}_{\tau g}(x^k - \tau \nabla f(x^k) - \tau D^*(2\tilde{y}^{k+1} - y^k)), \\ (x^{k+1}, y^{k+1}) = \rho_k(\tilde{x}^{k+1}, \tilde{y}^{k+1}) + (1 - \rho_k)(x^k, y^k) \end{cases} \quad (3.1)$$

Then, the corresponding algorithm is given below, called Algorithm 1.

Algorithm 1 A primal-dual splitting algorithm(PDS).

Initialization: Choose $x^0 \in \mathcal{X}$, $y^0 \in \mathcal{Y}$, relaxation parameters $(\rho_k)_{k \in \mathbb{N}}$, and proximal parameters $\sigma > 0$, $\tau > 0$.

Iterations ($k \geq 0$): Update x^k , y^k as follows

$$\begin{cases} \tilde{y}^{k+1} = \text{prox}_{\sigma h^*}(y^k + \sigma D x^k), \\ \tilde{x}^{k+1} = \text{prox}_{\tau g}(x^k - \tau \nabla f(x^k) - \tau D^*(2\tilde{y}^{k+1} - y^k)), \\ (x^{k+1}, y^{k+1}) = \rho_k(\tilde{x}^{k+1}, \tilde{y}^{k+1}) + (1 - \rho_k)(x^k, y^k). \end{cases}$$

end for

For Algorithm 1, the author given the following Theorem.

Theorem 3.1. ([1]) Let $\sigma > 0$, $\tau > 0$ and the sequences $(\rho_k)_{k \in \mathbb{N}}$, be the parameters of Algorithms 1. Let β be the Lipschitz constant and suppose that $\beta > 0$. Then the following hold:

- (i) $\frac{1}{\tau} - \sigma\|D\|^2 > 0$,
- (ii) $\forall k \in \mathbb{N}, \rho_k \in]0, \delta[$, where $\delta = 2 - \frac{\beta}{2}(\frac{1}{\tau} - \sigma\|D\|^2)^{-1} \in [1, 2[$,
- (iii) $\sum_{k \in \mathbb{N}} \rho_k(\delta - \rho_k) = +\infty$.

Let the sequences (x^k, y^k) be generated by Algorithms 1. Then the sequence $\{x_k\}$ converges to a solution of Problem (1.1).

The fixed point characterization provided by Condat [1] suggests solving Problem (1.1) via the fixed point iteration scheme (3.1) for a suitable value of the parameter $\sigma > 0, \tau > 0$. This iteration, which is referred to as a primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. A very natural idea is to provide a more general iteration in which the coefficient $\sigma > 0$ and $\tau > 0$ are made iteration-dependent to solve the general Problem (1.1), then we can obtain the following iteration scheme:

$$\begin{cases} \tilde{y}^{k+1} = \text{prox}_{\sigma_k h^*}(y^k + \sigma_k D x^k), \\ \tilde{x}^{k+1} = \text{prox}_{\tau_k g}(x^k - \tau_k \nabla f(x^k) - \tau_k D^*(2\tilde{y}^{k+1} - y^k)), \\ (x^{k+1}, y^{k+1}) = \rho_k(\tilde{x}^{k+1}, \tilde{y}^{k+1}) + (1 - \rho_k)(x^k, y^k) \end{cases} \quad (3.2)$$

which produces our proposed method Algorithm 3.2, described below. This algorithm can also be deduced from the fixed point formulation, whose detail we will give in the following. On the other hand, since the parameter $\sigma_k > 0$ and $\tau_k > 0$ are dynamic, so we call our method a primal-dual splitting algorithm with dynamic stepsize, and abbreviate it as PDSDS. If $\sigma_k \equiv \sigma$ and $\tau_k \equiv \tau$ then form (3.1) is equivalent to form (3.2). So PDS can be seen as a special case of PDSDS.

Algorithm 2 A primal-dual splitting algorithm with dynamic stepsize(PDSDS).

Initialization: Choose $x^0 \in \mathcal{X}, y^0 \in \mathcal{Y}$, relaxation parameters $(\rho_k)_{k \in \mathbb{N}}$, and proximal parameters $\liminf_{k \rightarrow \infty} \sigma_k > 0$, $\liminf_{k \rightarrow \infty} \tau_k > 0$.

Iterations ($k \geq 0$): Update x^k, y^k as follows

$$\begin{cases} \tilde{y}^{k+1} = \text{prox}_{\sigma_k h^*}(y^k + \sigma_k D x^k), \\ \tilde{x}^{k+1} = \text{prox}_{\tau_k g}(x^k - \tau_k \nabla f(x^k) - \tau_k D^*(2\tilde{y}^{k+1} - y^k)), \\ (x^{k+1}, y^{k+1}) = \rho_k(\tilde{x}^{k+1}, \tilde{y}^{k+1}) + (1 - \rho_k)(x^k, y^k) \end{cases}$$

end for

Now, we claim the convergence results for Algorithms 2.

Theorem 3.2. *Assume that the minimization Problem (1.1) is consistent, $\liminf_{k \rightarrow \infty} \sigma_k > 0$, and $\liminf_{k \rightarrow \infty} \tau_k > 0$. Let the sequences $(\rho_k)_{k \in \mathbb{N}}$, be the parameters of Algorithms 2. Let β be the Lipschitz constant and suppose that $\beta > 0$. Then the following hold:*

- (i) $\frac{1}{\liminf_{k \rightarrow \infty} \tau_k} - \liminf_{k \rightarrow \infty} \sigma_k \|D\|^2 > \frac{\beta}{2}$,
- (ii) $\forall k \in \mathbb{N}, \rho_k \in]0, \delta_k[$, where $\delta_k = 2 - \frac{\beta}{2}(\frac{1}{\tau_k} - \sigma_k \|D\|^2)^{-1} \in [1, 2[$,
- (iii) $0 < \liminf_{k \rightarrow \infty} \rho_k \leq \limsup_{k \rightarrow \infty} \rho_k < \limsup_{k \rightarrow \infty} \delta_k$ and $1 \leq \liminf_{k \rightarrow \infty} \delta_k \leq \limsup_{k \rightarrow \infty} \delta_k < 2$.

Let the sequences (x^k, y^k) be generated by Algorithms 2. Then the sequence $\{x_k\}$ converges to a solution of Problem (1.1).

We consider the case where D is injective (in particular, it is implicit that $\dim(\mathcal{X}) \leq \dim(\mathcal{Y})$). In the latter case, we denote by $\mathcal{R} = \text{Im}(D)$ the image of D and by D^{-1} the inverse of D on $\mathcal{R} \rightarrow \mathcal{X}$. We emphasize the fact that the inclusion $\mathcal{R} \subset \mathcal{Y}$ might be strict. We denote by ∇ the gradient operator. We make the following assumption:

Assumption 3.1. *The following facts holds true:*

- (1) D is injective;
- (2) $\nabla(f \circ D)^{-1}$ is L -Lipschitz continuous on \mathcal{R} .

For proximal parameters $\liminf_{k \rightarrow \infty} \mu_k > 0$, $\liminf_{k \rightarrow \infty} \tau_k > 0$, we consider the following algorithm which we shall refer to as ADMMDS⁺.

Algorithm 3 ADMMDS⁺.

Iterations ($k \geq 0$): Update x^k, u^k, y^k, z^k as follows

$$\begin{cases} z^{k+1} = \arg \min_{w \in \mathcal{Y}} [h(w) + \frac{\|w - (Dx^k + \mu_k y^k)\|^2}{2\mu_k}], & (a) \\ y^{k+1} = y^k + \mu_k^{-1}(Dx^k - z^{k+1}), & (b) \\ u^{k+1} = (1 - \tau_k \mu_k^{-1})Dx^k + \tau_k \mu_k^{-1}z^{k+1}, & (c) \\ x^{k+1} = \arg \min_{w \in \mathcal{X}} [g(w) + \langle \nabla f(x^k), w \rangle + \frac{\|Dw - u^{k+1} - \tau_k y^{k+1}\|^2}{2\tau_k}] & (d) \end{cases}$$

end for

Theorem 3.3. *Assume that the minimization Problem (1.1) is consistent, $\liminf_{k \rightarrow \infty} \mu_k > 0$, and $\liminf_{k \rightarrow \infty} \tau_k > 0$. Let Assumption 2.1 and Assumption 3.1 hold true and*

$\frac{1}{\liminf_{k \rightarrow \infty} \tau_k} - \frac{1}{\liminf_{k \rightarrow \infty} \mu_k} > \frac{L}{2}$. Let the sequences (x^k, y^k) be generated by Algorithms 3. Then the sequence $\{x_k\}$ converges to a solution of Problem (1.1).

3.2 Proofs of convergence

From the proof of Theorem 3.1 for Algorithm 1, we know that Algorithm 1 has the structure of a forward-backward iteration, when expressed in terms of nonexpansive operators on $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$, equipped with a particular inner product.

Let the inner product $\langle \cdot, \cdot \rangle_I$ in \mathcal{Z} be defined as

$$\langle z, z' \rangle := \langle x, x' \rangle + \langle y, y' \rangle, \quad \forall z = (x, y), z' = (x', y') \in \mathcal{Z}. \quad (3.3)$$

By endowing \mathcal{Z} with this inner product, we obtain the Euclidean space denoted by \mathcal{Z}_I . Let us define the bounded linear operator on \mathcal{Z} ,

$$P := \begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} \frac{1}{\tau} & -D^* \\ -D & \frac{1}{\sigma} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (3.4)$$

From the assumptions $\beta > 0$ and (i), we can easily check that P is positive definite. Hence, we can define another inner product $\langle \cdot, \cdot \rangle_P$ and norm $\|\cdot\|_P = \langle \cdot, \cdot \rangle_P^{\frac{1}{2}}$ in \mathcal{Z} as

$$\langle z, z' \rangle_P = \langle z, z' \rangle_I. \quad (3.5)$$

We denote by \mathcal{Z}_P the corresponding Euclidean space.

Lemma 3.1. ([1]). *Let the conditions (i)-(iv) in Theorem 3.1 be true. For every $n \in \mathbb{N}$, the following inclusion is satisfied by $\tilde{z}^{k+1} := (\tilde{x}^{k+1}, \tilde{y}^{k+1})$ computed by Algorithm 1:*

$$\tilde{z}^{k+1} := (I + P^{-1} \circ A)^{-1} \circ (I - P^{-1} \circ B)(z^k), \quad (3.6)$$

where

$$A := \begin{pmatrix} \partial g & D^* \\ -D & \partial h^* \end{pmatrix}, B := \begin{pmatrix} \nabla f \\ 0 \end{pmatrix}.$$

Set $M_1 = P^{-1} \circ A$, $M_2 = P^{-1} \circ B$, $T_1 = (I + M_1)^{-1}$, $T_2 = (I - M_2)^{-1}$, and $T = T_1 \circ T_2$. Then $T_1 \in \mathcal{A}(\mathcal{Z}_P, \frac{1}{2})$ and $T_2 \in \mathcal{A}(\mathcal{Z}_P, \frac{1}{2\kappa})$, $\kappa := (\frac{1}{\tau} - \sigma\|D\|^2)/\beta$. Then $T \in \mathcal{A}(\mathcal{Z}_P, \frac{1}{\delta})$ and $\delta = 2 - \frac{1}{2\kappa}$.

In association with Lemma 2.1 and Lemma 3.1, we obtained Theorem 3.1

Now, we are ready to prove Theorem 3.2

Proof. By setting

$$P_k := \begin{pmatrix} \frac{1}{\tau_k} & -D^* \\ -D & \frac{1}{\sigma_k} \end{pmatrix},$$

then the Algorithm 3.2 can be described as follows:

$$\tilde{z}^{k+1} := (I + P_k^{-1} \circ A)^{-1} \circ (I - P_k^{-1} \circ B)(z^k). \quad (3.7)$$

Considering the relaxation step, we obtain

$$z^{k+1} := \rho_k(I + P_k^{-1} \circ A)^{-1} \circ (I - P_k^{-1} \circ B)(z^k) + (1 - \rho_k)z^k. \quad (3.8)$$

Let $M_1^k = P_k^{-1} \circ A$, $M_2^k = P_k^{-1} \circ B$, $T_1^k = (I + M_1^k)^{-1}$, $T_2^k = (I - M_2^k)^{-1}$, and $T^k = T_1^k \circ T_2^k$. Then $T_1^k \in \mathcal{A}(\mathcal{Z}_P, \frac{1}{2})$ [3, Corollary 23.8].

First, let us prove the cocoercivity of M_2^k . Since the sequence τ_k is bounded, there exists a convergent subsequence converges to τ without loss of generality, we may assume that the convergent subsequence is τ_k itself, then we have $\frac{1}{\tau_k} \rightarrow \frac{1}{\tau}$, so $\forall \varepsilon > 0$, $\exists N_1$, such that when $n \geq N_1$, $\frac{1}{\tau_k} \geq \frac{1}{\tau} - \varepsilon$. With the same idea, for sequence σ_k , we also have $\sigma_k \rightarrow \sigma$, then for the above $\varepsilon \exists N_2$, such that when $n \geq N_2$, $\sigma_k \leq \sigma + \varepsilon$. Set $N_0 = \max\{N_1, N_2\}$, when $n \geq N_0$, we have $\frac{1}{\tau_k} \geq \frac{1}{\tau} - \varepsilon$, $\sigma_k \leq \sigma + \varepsilon$. Then for every $z = (x, y)$, $z' = (x', y') \in \mathcal{Z}$ and $\forall n \geq N_0$, we have

$$\begin{aligned} \|M_2^k(z) - M_2^k(z')\|_P^2 &= \frac{1}{(\frac{1}{\tau_k} - \sigma_k DD^*)} \|\nabla f(x) - \nabla f(x')\|^2 \\ &\quad + \frac{1}{(\frac{1}{\tau_k} - \sigma_k DD^*)^2} (\frac{1}{\tau} - \frac{1}{\tau_k}) \|\nabla f(x) - \nabla f(x')\|^2 \\ &\quad + \frac{\sigma_k D^2}{(\frac{1}{\tau_k} - \sigma_k DD^*)^2} (\frac{\sigma_k}{\sigma} - 1) \|\nabla f(x) - \nabla f(x')\|^2 \\ &\leq \frac{1}{(\frac{1}{\tau_k} - \sigma_k \|D\|^2)} \|\nabla f(x) - \nabla f(x')\|^2 \\ &\quad + \frac{\varepsilon}{(\frac{1}{\tau_k} - \sigma_k DD^*)^2} \|\nabla f(x) - \nabla f(x')\|^2 \end{aligned}$$

$$\begin{aligned}
& + \frac{\sigma_k D^2}{(\frac{1}{\tau_k} - \sigma_k D D^*)^2} \frac{\varepsilon}{\sigma} \|\nabla f(x) - \nabla f(x')\|^2 \\
& = \frac{1}{(\frac{1}{\tau_k} - \sigma_k D D^*)} \|\nabla f(x) - \nabla f(x')\|^2 \\
& + \frac{\varepsilon}{(\frac{1}{\tau_k} - \sigma_k D D^*)^2} (1 + \frac{\sigma_k}{\sigma} D^2) \|\nabla f(x) - \nabla f(x')\|^2,
\end{aligned}$$

by the arbitrariness of ε , we have

$$\begin{aligned}
\|M_2^k(z) - M_2^k(z')\|_P^2 & \leq \frac{1}{(\frac{1}{\tau_k} - \sigma_k \|D\|^2)} \|\nabla f(x) - \nabla f(x')\|^2 \\
& = \frac{1}{\pi_k \beta} \|\nabla f(x) - \nabla f(x')\|^2 \\
& \leq \frac{\beta}{\pi_k} \|x - x'\|^2,
\end{aligned} \tag{3.9}$$

where $\pi_k = (\frac{1}{\tau_k} - \sigma_k \|D\|^2)/\beta$. We define the linear operator $Q : (x, y) \rightarrow (x, 0)$ of \mathcal{Z} . Since $P - \beta\pi_k Q$ is positive in \mathcal{Z}_I , we have

$$\begin{aligned}
\beta\pi_k \|x - x'\|^2 & = \beta\pi_k \langle (z - z'), Q(z - z') \rangle_I \\
& \leq \langle (z - z'), P(z - z') \rangle_I = \|z - z'\|_P^2.
\end{aligned} \tag{3.10}$$

Putting together (3.9) and (3.10), we get

$$\pi_k \|M_2^k(z) - M_2^k(z')\|_P \leq \|z - z'\|_P^2. \tag{3.11}$$

So that $\pi_k M_2^k$ is nonexpansive in \mathcal{Z}_P . Let us define on \mathcal{Z}_P the function $J : (x, y) \rightarrow P_k^{-1}f(x)$. Then, in \mathcal{Z}_P , $\nabla J = M_2^k$. Therefore, from Lemma 2.2, $\pi_k M_2^k$ is firmly nonexpansive in \mathcal{Z}_P . Then $T_2^k \in \mathcal{A}(\mathcal{Z}_P, \frac{1}{2\pi_k})$ [3, Proposition 4.33]. Hence, from Lemma 2.3, we know $T^k \in \mathcal{A}(\mathcal{Z}_P, \frac{1}{\delta_k})$, and $\delta_k = 2 - \frac{1}{2\pi_k}$.

Next, we will prove the convergence of Algorithm 2.

Since for each n , T^k is $\frac{1}{\delta_k}$ -averaged. Therefore, we can write

$$T^k = (1 - \frac{1}{\delta_k})I + \frac{1}{\delta_k}S^k, \tag{3.12}$$

where S^k is nonexpansive and $\frac{1}{\delta_k} \in]\frac{1}{2}, 1]$. Then we can rewrite (3.8) as

$$z^{k+1} = (1 - \frac{\rho_k}{\delta_k})z^k + \frac{\rho_k}{\delta_k}S^k z^k = (1 - \alpha_k)z^k + \alpha_k S^k z^k, \tag{3.13}$$

where $\alpha_k = \frac{\rho_k}{\delta_k}$. Let $\hat{z} \in \text{Fix}(S)$, where $\hat{z} = (\hat{x}, \hat{y})$, then \hat{x} is a solution of (1.1), noticing that $S^k \hat{z} = \hat{z}$, we have

$$\begin{aligned} \|z^{k+1} - \hat{z}\|_P^2 &= (1 - \alpha_k) \|z^k - \hat{z}\|_P^2 + \alpha_k \|S^k z^k - \hat{z}\|_P^2 - \alpha_k (1 - \alpha_k) \|z^k - S^k z^k\|_P^2 \\ &\leq \|z^k - \hat{z}\|_P^2 - \alpha_k (1 - \alpha_k) \|z^k - S^k z^k\|_P^2. \end{aligned} \quad (3.14)$$

Which implies that

$$\|z^{k+1} - \hat{z}\|_P^2 \leq \|z^k - \hat{z}\|_P^2. \quad (3.15)$$

This implies that sequence $\{z^k\}_{k=0}^\infty$ is a Fejér monotone sequence, and $\lim_{k \rightarrow \infty} \|z^{k+1} - \hat{z}\|_P$ exists.

From the condition (iii) of Theorem 3.1, it is easy to find that

$$0 < \liminf_{k \rightarrow \infty} \alpha_k \leq \limsup_{k \rightarrow \infty} \alpha_k < 1.$$

Therefor, there exists $\underline{a}, \bar{a} \in (0, 1)$ such that $\underline{a} < \alpha_k < \bar{a}$. By (3.14), we know

$$\begin{aligned} \underline{a}(1 - \bar{a}) \|z^k - S^k z^k\|_P^2 &\leq \alpha_k (1 - \alpha_k) \|z^k - S^k z^k\|_P^2 \\ &\leq \|z^k - \hat{z}\|_P^2 - \|z^{k+1} - \hat{z}\|_P^2. \end{aligned}$$

Hence

$$\lim_{k \rightarrow \infty} \|z^k - S^k z^k\|_P = 0. \quad (3.16)$$

Since the sequence $\{z^k\}$ is bounded and there exists a convergent subsequence $\{z^{k_j}\}$ such that

$$z^{k_j} \rightarrow \tilde{z}, \quad (3.17)$$

for some $\tilde{z} \in \mathcal{X} \times \mathcal{Y}$.

From (3.14), we have

$$\lim_{j \rightarrow \infty} \|z^{k_j} - S^{k_j} z^{k_j}\|_P = 0. \quad (3.18)$$

Since the sequence τ_k is bounded, there exists a subsequence $\tau_{k_j} \subset \tau_k$ such that $\frac{1}{\tau_{k_j}} \rightarrow \frac{1}{\tau}$. With the same idea, we have $\sigma_{k_j} \rightarrow \sigma$. Then we obtain that $\delta = 2 - \frac{1}{2\pi} \in [1, 2[$. Therefor, we know that $T = (I + P^{-1} \circ A)^{-1} \circ (I - P^{-1} \circ B)$ is $\frac{1}{\delta}$ -averaged. So there exists a nonexpansive mapping S such that

$$T = (I + P^{-1} \circ A)^{-1} \circ (I - P^{-1} \circ B) = (1 - \frac{1}{\delta})I + \frac{1}{\delta}S,$$

where $\delta_{k_j} \rightarrow \delta$. Because the solution of the Problem (1.1) is consistent, we know that $\bigcap_{k=1}^{\infty} \text{Fix}(S^k) = \text{Fix}(S) \neq \emptyset$. Then we will prove $\lim_{j \rightarrow \infty} \|z^{k_j} - Sz^{k_j}\|_P = 0$. In fact, we have

$$\begin{aligned}
\|z^{k_j} - Sz^{k_j}\|_P &\leq \|z^{k_j} - S^{k_j}z^{k_j}\|_P + \|S^{k_j}z^{k_j} - Sz^{k_j}\|_P \\
&= \|z^{k_j} - S^{k_j}z^{k_j}\|_P + \|(1 - \delta_{k_j})z^{k_j} + \delta_{k_j}T^{k_j}z^{k_j} - (1 - \delta)z^{k_j} - \delta Tz^{k_j}\|_P \\
&\leq \|z^{k_j} - S^{k_j}z^{k_j}\|_P + |\delta_{k_j} - \delta|(\|z^{k_j}\|_P + \|Tz^{k_j}\|_P) + \delta\|T^{k_j}z^{k_j} - Tz^{k_j}\|_P.
\end{aligned} \tag{3.19}$$

Since $(I + P_{k_j}^{-1} \circ A)^{-1} = J_{P_{k_j}^{-1}A}$, $(I + P^{-1} \circ A)^{-1} = J_{P^{-1}A}$, so from Lemma 2.5, we know that

$$\begin{aligned}
\|T^{k_j}z^{k_j} - Tz^{k_j}\|_P &= \|(I + P_{k_j}^{-1} \circ A)^{-1} \circ (I - P_{k_j}^{-1} \circ B)z^{k_j} \\
&\quad - (I + P^{-1} \circ A)^{-1} \circ (I - P^{-1} \circ B)z^{k_j}\|_P \\
&\leq \|J_{P_{k_j}^{-1}A} \circ (I - P_{k_j}^{-1} \circ B)z^{k_j} - J_{P_{k_j}^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j}\|_P \\
&\quad + \|J_{P_{k_j}^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j} - J_{P^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j}\|_P \\
&\leq |P_{k_j}^{-1} - P^{-1}|\|Bz^{k_j}\|_P + \|J_{P^{-1}A}(\frac{P^{-1}}{P_{k_j}^{-1}}(I - P^{-1} \circ B)z^{k_j} \\
&\quad + (1 - \frac{P^{-1}}{P_{k_j}^{-1}})J_{P_{k_j}^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j}) - J_{P^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j}\|_P \\
&\leq |P_{k_j}^{-1} - P^{-1}|\|Bz^{k_j}\|_P + \|\frac{P^{-1}}{P_{k_j}^{-1}}(I - P^{-1} \circ B)z^{k_j} \\
&\quad + (1 - \frac{P^{-1}}{P_{k_j}^{-1}})J_{P_{k_j}^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j} - (I - P^{-1} \circ B)z^{k_j}\|_P \\
&\leq |P_{k_j}^{-1} - P^{-1}|\|Bz^{k_j}\|_P + |1 - \frac{P^{-1}}{P_{k_j}^{-1}}|\|J_{P_{k_j}^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j} \\
&\quad - (I - P^{-1} \circ B)z^{k_j}\|_P.
\end{aligned} \tag{3.20}$$

Put (3.20) into (3.19), we obtain that

$$\begin{aligned}
\|z^{k_j} - Sz^{k_j}\|_P &\leq \|z^{k_j} - S^{k_j}z^{k_j}\|_P + |\delta_{k_j} - \delta|(\|z^{k_j}\|_P + \|Tz^{k_j}\|_P) \\
&\quad + \delta|P_{k_j}^{-1} - P^{-1}|\|Bz^{k_j}\|_P + \delta|1 - \frac{P^{-1}}{P_{k_j}^{-1}}|\|J_{P_{k_j}^{-1}A} \circ (I - P^{-1} \circ B)z^{k_j}
\end{aligned}$$

$$- (I - P^{-1} \circ B)z^{k_j}\|_P. \quad (3.21)$$

From $\delta_{k_j} \rightarrow \delta$, $P_{k_j}^{-1} \rightarrow P^{-1}$ and (3.18), we have

$$\lim_{j \rightarrow \infty} \|z^{k_j} - Sz^{k_j}\|_P = 0. \quad (3.22)$$

By Lemma 2.4, we know $\tilde{z} \in \text{Fix}(S)$. Moreover, we know that $\{\|z^k - \hat{z}\|_P\}$ is non-increasing for any fixed point \hat{z} of S . In particular, by choosing $\hat{z} = \tilde{z}$, we have $\{\|z^k - \tilde{z}\|_P\}$ is non-increasing. Combining this and (3.17) yields

$$\lim_{k \rightarrow \infty} z^k = \tilde{z}. \quad (3.23)$$

Writing $\tilde{z} = (\tilde{x}, \tilde{y})$, then we have \tilde{x} is the solution of Problem (1.1). □

Proof of Theorem 3.3 for Algorithm 3. Before providing the proof of Theorem 3.3, let us introduce the following notation and Lemma.

Lemma 3.2. *Given a Euclidean space \mathcal{E} , consider the minimization problem $\min_{\lambda \in \mathcal{E}} \bar{f}(\lambda) + \bar{g}(\lambda) + h(\lambda)$, where $\bar{g}, h \in \Gamma_0(\mathcal{E})$ and where \bar{f} is convex and differentiable on \mathcal{E} with a L -Lipschitz continuous gradient. Assume that the infimum is attained and that $0 \in \text{ri}(\text{dom}h - \text{dom}\bar{g})$. Let $\liminf_{k \rightarrow \infty} \mu_k > 0$, $\liminf_{k \rightarrow \infty} \tau_k > 0$ be such that $\frac{1}{\liminf_{k \rightarrow \infty} \tau_k} - \frac{1}{\liminf_{k \rightarrow \infty} \sigma_k} > \frac{L}{2}$, and consider the iterates*

$$\begin{cases} y^{k+1} = \text{prox}_{\mu_k^{-1}h^*}(y^k + \mu_k^{-1}\lambda^k), \\ \lambda^{k+1} = \text{prox}_{\tau_k\bar{g}}(\lambda^k - \tau_k\nabla\bar{f}(\lambda^k) - \tau_k(2y^{k+1} - y^k)). \end{cases} \quad (3.24a)$$

Then for any initial value $(\lambda^0, y^0) \in \mathcal{E} \times \mathcal{E}$, the sequence (λ^k, y^k) converges to a primal-dual point $(\tilde{\lambda}, \tilde{y})$, i.e., a solution of the equation

$$\min_{\lambda \in \mathcal{E}} \bar{f}(\lambda) + \bar{g}(\lambda) + h(\lambda) = -\min_{y \in \mathcal{E}} (\bar{f} + \bar{g})^*(y) + h^*(y). \quad (3.25)$$

Proof. It is easy to see that the Lemma 3.2 is a special case of Theorem 3.2. So we can obtain Lemma 3.2 from Theorem 3.2 directly. □

Elaborating on Lemma 3.2, we are now ready to establish the Theorem 3.3.

By setting $\mathcal{E} = \mathcal{S}$ and by assuming that \mathcal{E} is equipped with the same inner product as \mathcal{Y} , one can notice that the functions $\bar{f} = f \circ D^{-1}$, $\bar{g} = g \circ D^{-1}$ and h satisfy the conditions of Lemma 3.2. Moreover, since $(\bar{f} + \bar{g})^* = (f + g)^* \circ D^*$, one can also notice that (\tilde{x}, \tilde{y}) is a primal-dual point associated with Eq. (2.1) if and only if $(D\tilde{x}, \tilde{y})$ is a primal-dual point associated with Eq. (3.25). With the same idea for the proof of Theorem 1 of [2], we can recover the ADMMDS⁺ from the iterations (3.24).

3.3 Connections to other algorithms

We will further establish the connections to other existing methods.

When $\mu_k \equiv \mu$ and $\tau_k \equiv \tau$, the ADMMDS⁺ boils down to the ADMM⁺ whose iterations are given by:

$$\begin{cases} z^{k+1} = \operatorname{argmin}_{w \in \mathcal{Y}} [h(w) + \frac{\|w - (Dx^k + \mu y^k)\|^2}{2\mu}], \\ y^{k+1} = y^k + \mu^{-1}(Dx^k - z^{k+1}), \\ u^{k+1} = (1 - \tau\mu^{-1})Dx^k + \tau\mu^{-1}z^{k+1}, \\ x^{k+1} = \operatorname{argmin}_{w \in \mathcal{X}} [g(w) + \langle \nabla f(x^k), w \rangle + \frac{\|Dw - u^{k+1} - \tau y^{k+1}\|^2}{2\tau}]. \end{cases}$$

In the special case $h \equiv 0$, $D = I$, $\mu_k \equiv \mu$ and $\tau_k \equiv \tau$ it can be easily verified that y^k is null for all $k \geq 1$ and $u^k = x^k$. Then, the ADMMDS⁺ boils down to the standard Forward-Backward algorithm whose iterations are given by:

$$\begin{aligned} x^{k+1} &= \operatorname{argmin}_{w \in \mathcal{X}} g(w) + \frac{1}{2\tau} \|w - (x^k - \tau \nabla f(x^k))\|^2 \\ &= \operatorname{prox}_{\tau g}(x^k - \tau \nabla f(x^k)). \end{aligned}$$

One can remark that μ has disappeared thus it can be set as large as wanted so the condition on stepsize τ from Theorem 3.3 boils down to $\tau < 2/L$. Applications of this algorithm with particular functions appear in well known learning methods such as ISTA [11].

4 Coordinate descent

4.1 Randomized krasnosel'skii-mann iterations

Consider the space $\mathcal{Z} = \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_J$ for some $J \in \mathbb{N}^*$ where for any j , \mathcal{Z}_j is a Euclidean space. For \mathcal{Z} equipped with the scalar product $\langle x, y \rangle = \sum_{j=1}^J \langle x_j, y_j \rangle_{\mathcal{Z}_j}$ where $\langle \cdot, \cdot \rangle_{\mathcal{Z}_j}$ is the scalar product in \mathcal{Z}_j . For $j \in \{1, \dots, J\}$, let $T_j : \mathcal{Z} \rightarrow \mathcal{Z}_j$ be the components of the output of operator $T : \mathcal{Z} \rightarrow \mathcal{Z}$ corresponding to \mathcal{Z}_j , so, we have $Tx = (T_1x, \dots, T_Jx)$. Let $2^{\mathcal{J}}$ be the power set of $\mathcal{J} = \{1, \dots, J\}$. For any $\vartheta \in 2^{\mathcal{J}}$, we denote the operator $\hat{T}^\vartheta : \mathcal{Z} \rightarrow \mathcal{Z}$ by $\hat{T}_j^\vartheta x = T_jx$ for $j \in \vartheta$ and $\hat{T}_j^\vartheta x = x_j$ for otherwise. On some probability space $(\Omega, \mathcal{F}, \mathbb{P})$, we introduce a random i.i.d. sequence $(\zeta^k)_{k \in \mathbb{N}^*}$ such that $\zeta^k : \Omega \rightarrow 2^{\mathcal{J}}$ i.e. $\zeta^k(\omega)$ is a subset of \mathcal{J} . Assume that the following holds:

$$\forall j \in \mathcal{J}, \exists \vartheta \in 2^{\mathcal{J}}, j \in \vartheta \quad \text{and} \quad \mathbb{P}(\zeta_1 = \vartheta) > 0. \quad (4.1)$$

Lemma 4.1. *(Theorem 3 of [2]). Let $T : \mathcal{Z} \rightarrow \mathcal{Z}$ be η -averaged and $\text{Fix}(T) \neq \emptyset$. Let $(\zeta^k)_{k \in \mathbb{N}^*}$ be a random i.i.d. sequence on $2^{\mathcal{J}}$ such that Condition (4.1) holds. If for all k , sequence $(\beta_k)_{k \in \mathbb{N}}$ satisfies*

$$0 < \liminf_{k \rightarrow \infty} \beta_k \leq \limsup_{k \rightarrow \infty} \beta_k < \frac{1}{\eta}.$$

Then, almost surely, the iterated sequence

$$x^{k+1} = x^k + \beta_k (\hat{T}^{\zeta^{k+1}} x^k - x^k) \quad (4.2)$$

converges to some point in $\text{Fix}(T)$.

4.2 Randomized Modified krasnosel'skii- mann iterations

Theorem 4.1. *Let T be η -averaged and T^k be η_k -averaged on \mathcal{Z} and $\bigcap_{k=1}^{\infty} \text{Fix}(T^k) = \text{Fix}(T) \neq \emptyset$, and $T^k \rightarrow T$. Let $(\zeta^k)_{k \in \mathbb{N}^*}$ be a random i.i.d. sequence on $2^{\mathcal{J}}$ such that Condition (4.1) holds. If for all k , sequence $(\beta_k)_{k \in \mathbb{N}}$ satisfies*

$$0 < \beta_k < \frac{1}{\eta_k}, 0 < \liminf_{k \rightarrow \infty} \beta_k \leq \limsup_{k \rightarrow \infty} \beta_k < \frac{1}{\limsup_{k \rightarrow \infty} \eta_k}.$$

Then, almost surely, the iterated sequence

$$x^{k+1} = x^k + \beta_k(\hat{T}^{k,(\zeta^{k+1})}x^k - x^k) \quad (4.3)$$

converges to some point in $\text{Fix}(T)$.

Proof. Define the operator $U^k = (1 - \beta_k)I + \beta_k T^k$; similarly, define $U^{k,(\vartheta)} = (1 - \beta_k)I + \beta_k T^{k,(\vartheta)}$. Observing that the operator U^k is $(\beta_k \eta_k)$ -averaged. From (4.3), we can know that $x^{k+1} = U^{k,(\zeta^{k+1})}x^k$. Set $p_\vartheta = \mathbb{P}(\zeta_1 = \vartheta)$ for any $\vartheta \in 2^{\mathcal{J}}$. Denote by $\|x\|^2 = \langle x, x \rangle$ the squared norm in \mathcal{Z} . Define a new inner product $x \bullet y = \sum_{j=1}^J q_j \langle x_j, y_j \rangle_j$ on \mathcal{Z} where $q_j^{-1} = \sum_{\vartheta \in 2^{\mathcal{J}}} p_\vartheta \mathbf{1}_{\{j \in \vartheta\}}$ and let $\|x\|^2 = x \bullet x$ be its associated squared norm. Consider any $x^* \in \text{Fix}(T)$. Conditionally to the sigma-field $\mathcal{F}^k = \sigma(\zeta_1, \dots, \zeta^k)$ we have

$$\begin{aligned} \mathbb{E}[\|x^{k+1} - \tilde{x}\|^2 | \mathcal{F}^k] &= \sum_{\vartheta \in 2^{\mathcal{J}}} p_\vartheta \|U^{k,(\vartheta)}x^k - \tilde{x}\|^2 \\ &= \sum_{\vartheta \in 2^{\mathcal{J}}} p_\vartheta \sum_{j \in \vartheta} q_j \|U_j^k x^k - \tilde{x}_j\|^2 + \sum_{\vartheta \in 2^{\mathcal{J}}} p_\vartheta \sum_{j \notin \vartheta} q_j \|x_j^k - \tilde{x}_j\|^2 \\ &= \|x^k - \tilde{x}\|^2 + \sum_{\vartheta \in 2^{\mathcal{J}}} p_\vartheta \sum_{j \in \vartheta} q_j (\|U_j^k x^k - \tilde{x}_j\|^2 - \|x_j^k - \tilde{x}_j\|^2) \\ &= \|x^k - \tilde{x}\|^2 + \sum_{j=1}^J (\|U_j^k x^k - \tilde{x}_j\|^2 - \|x_j^k - \tilde{x}_j\|^2) \\ &= \|x^k - \tilde{x}\|^2 + (\|U^k x^k - \tilde{x}\|^2 - \|x^k - \tilde{x}\|^2) \end{aligned}$$

Since U^k is $(\beta_k \eta_k)$ -averaged and that \tilde{x} is a fixed point of U^k , the term enclosed in the parentheses is no larger than $-\frac{1-\beta_k \eta_k}{\beta_k \eta_k} \|(I - U^k)x^k\|^2$. By $I - U^k = \beta_k(I - T^k)$, we have:

$$\begin{aligned} \mathbb{E}[\|x^{k+1} - \tilde{x}\|^2 | \mathcal{F}^k] &\leq \|x^k - \tilde{x}\|^2 - \beta_k(1 - \beta_k \eta_k) \|(I - T^k)x^k\|^2 \\ &\leq \|x^k - \tilde{x}\|^2 - \beta_k \eta_k (1 - \beta_k \eta_k) \|(I - T^k)x^k\|^2, \end{aligned} \quad (4.4)$$

which shows that $\|x^k - \tilde{x}\|^2$ is a nonnegative supermartingale with respect to the filtration (\mathcal{F}^k) . As such, it converges with probability one towards a random variable that is finite almost everywhere.

Given a countable dense subset Z of $\text{Fix}(T)$, there is a probability one set on which $\|x^k - x\| \rightarrow X_x \in [0, \infty)$ for all $x \in Z$. Let $x \in \text{Fix}(T)$, let $\varepsilon > 0$, and choose $x \in Z$

such that $\|\tilde{x} - x\| \leq \varepsilon$. With probability one, we have

$$\|x^k - \tilde{x}\| \leq \|x^k - x\| + \|\tilde{x} - x\| \leq X_x + 2\varepsilon,$$

for k large enough. Similarly $\|x^k - \tilde{x}\| \geq X_x - 2\varepsilon$, for k large enough. Therefore, we have

A₁: There is a probability one set on which $\|x^k - \tilde{x}\|$ converges for every $\tilde{x} \in \text{Fix}(T)$.

From the assumption on $(\beta_k)_{k \in \mathbb{N}}$, we know that $0 < \liminf_{k \rightarrow \infty} \beta_k \eta_k \leq \limsup_{k \rightarrow \infty} \beta_k \eta_k < 1$. So there exists $\bar{a}, \underline{a} \in (0, 1)$, such that $\underline{a} < \beta_k \eta_k < \bar{a}$. From (4.4), we have

$$\begin{aligned} \underline{a}(1 - \bar{a})\|(I - T^k)x^k\|^2 &\leq \alpha_k \beta_k (1 - \alpha_k \beta_k) \|(I - T^k)x^k\|^2 \\ &\leq \|x^k - \tilde{x}\|^2 - \mathbb{E}[\|x^{k+1} - \tilde{x}\|^2 | \mathcal{F}^k]. \end{aligned} \quad (4.5)$$

Taking the expectations on both sides of inequality (4.5) and iterating over k , we obtain

$$\mathbb{E}\|(I - T^k)x^k\|^2 \leq \frac{1}{\underline{a}(1 - \bar{a})}(x^0 - \tilde{x})^2.$$

By Markov's inequality and Borel Cantelli's lemma, we therefore obtain:

A₂: $(I - T^k)x^k \rightarrow 0$ almost surely.

We now consider an elementary event in the probability one set where **A₁** and **A₂** hold. On this event, since the sequence $(x^k)_{k \in \mathbb{N}}$ is bounded, so there exists a convergent subsequence $(x^{k_j})_{j \in \mathbb{N}}$ such that

$$\lim_{j \rightarrow \infty} \|x^{k_j} - \hat{x}\| = 0, \quad (4.6)$$

for some $\hat{x} \in \mathcal{Z}$.

From **A₂** and the condition $T^k \rightarrow T$, we have

$$\begin{aligned} \|x^{k_j} - Tx^{k_j}\| &\leq \|x^{k_j} - T^{k_j}x^{k_j}\| + \|T^{k_j}x^{k_j} - Tx^{k_j}\| \\ &\leq \|x^{k_j} - T^{k_j}x^{k_j}\| + \|T^{k_j} - T\|\|x^{k_j}\| \rightarrow 0. \end{aligned} \quad (4.7)$$

It then follows from Lemma 2.4 that $\hat{x} \in \text{Fix}(T)$. Moreover, we know that on this event, $\|x^k - \tilde{x}\|$ converges for any $\tilde{x} \in \text{Fix}(T)$. In particular, by choosing $\tilde{x} = \hat{x}$, we see that $\|x^k - \hat{x}\|$ converges. Combining this and (4.6) yields

$$\lim_{k \rightarrow \infty} \|x^k - \hat{x}\| = 0.$$

□

From Theorem 3.3, we know that the ADMMDS⁺ iterates are generated by the action of a η_k -averaged operator. Theorem 4.1 shows then that a stochastic coordinate descent version of the ADMMDS⁺ converges towards a primal-dual point. This result will be exploited in two directions: first, we describe a stochastic minibatch algorithm, where a large dataset is randomly split into smaller chunks. Second, we develop an asynchronous version of the ADMMDS⁺ in the context where it is distributed on a graph.

5 Application to stochastic approximation

5.1 Problem setting

Given an integer $N > 1$, consider the problem of minimizing a sum of composite functions

$$\inf_{x \in \mathcal{X}} \sum_{n=1}^N (f_n(x) + g_n(x)), \quad (5.1)$$

where we make the following assumption:

Assumption 5.1. *For each $n = 1, \dots, N$,*

- (1) f_n is a convex differentiable function on \mathcal{X} , and its gradient ∇f_n is $1/\beta$ -Lipschitz continuous on \mathcal{X} for some $\beta \in (0, +\infty)$;*
- (2) $g_n \in \Gamma_0(\mathcal{X})$;*
- (3) The infimum of Problem (5.1) is attained;*
- (4) $\cap_{n=1}^N \text{ri dom } g_n \neq \emptyset$.*

This problem arises for instance in large-scale learning applications where the learning set is too large to be handled as a single block. Stochastic minibatch approaches consist in splitting the data set into N chunks and to process each chunk in some order, one at a time. The quantity $f_n(x) + g_n(x)$ measures the inadequacy between the model (represented by parameter x) and the n -th chunk of data. Typically, f_n stands for a data fitting term whereas g_n is a regularization term which penalizes the occurrence of erratic solutions. As an example, the case where f_n is quadratic and g_n is the l_1 -norm

reduces to the popular LASSO problem [12]. In particular, it is also useful to recover sparse signal.

5.2 Instantiating the ADMMDS⁺

We regard our stochastic minibatch algorithm as an instance of the ADMMDS⁺ coupled with a randomized coordinate descent. In order to end that, we rephrase Problem (5.1) as

$$\inf_{x \in \mathcal{X}^N} \sum_{n=1}^N (f_n(x) + g_n(x)) + \iota_{\mathcal{C}}(x), \quad (5.2)$$

where the notation x_n represents the n -th component of any $x \in \mathcal{X}^N$, \mathcal{C} is the space of vectors $x \in \mathcal{X}^N$ such that $x_1 = \dots = x_N$. On the space \mathcal{X}^N , we set $f(x) = \sum_n f_n(x_n)$, $g(x) = \sum_n g_n(x_n)$, $h(x) = \iota_{\mathcal{C}}$ and $D = I_{\mathcal{X}^N}$ the identity matrix. Problem (5.2) is equivalent to

$$\min_{x \in \mathcal{X}^N} f(x) + g(x) + (h \circ D)(x). \quad (5.3)$$

We define the natural scalar product on \mathcal{X}^N as $\langle x, y \rangle = \sum_{n=1}^N \langle x_n, y_n \rangle$. Applying the ADMMDS⁺ to solve Problem (5.3) leads to the following iterative scheme:

$$\begin{aligned} z^{k+1} &= \text{proj}_{\mathcal{C}} \|x^k + \mu_k y^k\|^2, \\ y_n^{k+1} &= y_n^k + \mu_k^{-1} (x_n^k - z_n^{k+1}), \\ u_n^{k+1} &= (1 - \tau_k \mu_k^{-1}) x_n^k + \tau_k \mu_k^{-1} z_n^{k+1}, \\ x_n^{k+1} &= \arg \min_{w \in \mathcal{X}} [g_n(w) + \langle \nabla f_n(x^k), w \rangle + \frac{\|w - u_n^{k+1} - \tau_k y_n^{k+1}\|^2}{2\tau_k}], \end{aligned}$$

where $\text{proj}_{\mathcal{C}}$ is the orthogonal projection onto \mathcal{C} . Observe that for any $x \in \mathcal{X}^N$, $\text{proj}_{\mathcal{C}}(x)$ is equivalent to $(\bar{x}, \dots, \bar{x})$ where \bar{x} is the average of vector x , that is $\bar{x} = N^{-1} \sum_n x_n$. Consequently, the components of z^{k+1} are equal and coincide with $\bar{x}^k + \mu_k \bar{y}^k$ where \bar{x}^k and \bar{y}^k are the averages of x^k and y^k respectively. By inspecting the y^k n -update equation above, we notice that the latter equality simplifies even further by noting that $\bar{y}^{k+1} = 0$ or, equivalently, $\bar{y}^k = 0$ for all $k \geq 1$ if the algorithm is started with $\bar{y}^0 = 0$. Finally, for any n and $k \geq 1$, the above iterations reduce to

$$\bar{x}^k = \frac{1}{N} \sum_{n=1}^N x_n^k,$$

$$\begin{aligned}
y_n^{k+1} &= y_n^k + \mu_k^{-1}(x_n^k - \bar{x}^k), \\
u_n^{k+1} &= (1 - \tau_k \mu_k^{-1})x_n^k + \tau_k \mu_k^{-1} \bar{x}^k, \\
x_n^{k+1} &= \text{prox}_{\tau_k g_n}[u_n^{k+1} - \tau_k(\nabla f_n(x_n^k) + y_n^{k+1})].
\end{aligned}$$

These iterations can be written more compactly as

Algorithm 4 Minibatch ADMMSD⁺.

Initialization: Choose $x^0 \in \mathcal{X}$, $y^0 \in \mathcal{Y}$, s.t. $\sum_n v_n^0 = 0$.

Do

- $\bar{x}^k = \frac{1}{N} \sum_{n=1}^N x_n^k$,
 - For batches $n = 1, \dots, N$, do

$$\begin{aligned}
y_n^{k+1} &= y_n^k + \mu_k^{-1}(x_n^k - \bar{x}^k), \\
x_n^{k+1} &= \text{prox}_{\tau_k g_n}[(1 - 2\tau_k \mu_k^{-1})x_n^k - \tau_k \nabla f_n(x_n^k) + 2\tau_k \mu_k^{-1} \bar{x}^k - \tau_k y_n^k].
\end{aligned} \tag{5.4}$$
 - Increment k .
-

The following result is a straightforward consequence of Theorem 3.3.

Theorem 5.1. *Assume that the minimization Problem (5.3) is consistent, $\liminf_{k \rightarrow \infty} \mu_k > 0$, and $\liminf_{k \rightarrow \infty} \tau_k > 0$. Let Assumption 5.1 hold true and $\frac{1}{\liminf_{k \rightarrow \infty} \tau_k} - \frac{1}{\liminf_{k \rightarrow \infty} \mu_k} > \frac{L}{2}$. Let the sequences (\bar{x}^k, y^k) be generated by Minibatch ADMMSD⁺. Then for any initial point (x^0, y^0) such that $\bar{y}^0 = 0$, the sequence $\{\bar{x}^k\}$ converges to a solution of Problem (5.3).*

At each step k , the iterations given above involve the whole set of functions $f_n, g_n (n = 1, \dots, N)$. Our aim is now to propose an algorithm which involves a single couple of functions (f_n, g_n) per iteration.

5.3 A stochastic minibatch primal-dual splitting algorithm with dynamic stepsize

We are now in position to state the main algorithm of this section. The proposed stochastic minibatch primal-dual splitting algorithm with dynamic stepsize (SMPDSDS) is obtained upon applying the randomized coordinate descent on the minibatch ADMMSD⁺:

Algorithm 5 SMPDSDS.

Initialization: Choose $x^0 \in \mathcal{X}$, $y^0 \in \mathcal{Y}$.

Do

- Define $\bar{x}^k = \frac{1}{N} \sum_{n=1}^N x_n^k$, $\bar{y}^k = \frac{1}{N} \sum_{n=1}^N y_n^k$,
 - Pick up the value of ζ^{k+1} ,
 - For batch $n = \zeta^{k+1}$, set

$$y_n^{k+1} = y_n^k - \bar{y}^k + \frac{(x_n^k - \bar{x}^k)}{\mu_k}, \quad (5.5a)$$

$$x_n^{k+1} = \text{prox}_{\tau_k g_n}[(1 - 2\tau_k \mu_k^{-1})x_n^k - \tau_k \nabla f_n(x_n^k) - \tau_k y_n^k + 2\tau_k(\mu_k^{-1}\bar{x}^k + \bar{y}^k)]. \quad (5.5b)$$
 - For all batches $n \neq \zeta^{k+1}$, $y_n^{k+1} = y_n^k$, $x_n^{k+1} = x_n^k$.
 - Increment k .
-

Assumption 5.2. The random sequence $(\zeta^k)_{k \in \mathbb{N}^*}$ is i.i.d. and satisfies $\mathbb{P}[\zeta^1 = n] > 0$ for all $n = 1, \dots, N$.

Theorem 5.2. Assume that the minimization Problem (5.3) is consistent, $\liminf_{k \rightarrow \infty} \mu_k > 0$, and $\liminf_{k \rightarrow \infty} \tau_k > 0$. Let Assumption 5.1 and Assumption 5.2 hold true and $\frac{1}{\liminf_{k \rightarrow \infty} \tau_k} - \frac{1}{\liminf_{k \rightarrow \infty} \mu_k} > \frac{L}{2}$. Then for any initial point (x^0, y^0) , the sequence $\{\bar{x}^k\}$ generated by SMPDSDS algorithm converges to a solution of Problem (5.3).

Proof. Let us define $(\bar{f}, \bar{g}, h, D) = (f, g, h, I_{x^N})$ where the functions f , g , and h are the ones defined in section 5.2. Then the iterates $((y_n^{k+1})_{n=1}^N, (x_n^{k+1})_{n=1}^N)$ described by Equations (5.4) coincide with the iterates (y^{k+1}, x^{k+1}) described by Equations (3.24). If we write these equations more compactly as $(y^{k+1}, x^{k+1}) = T^k(y^k, x^k)$ where the operator T^k acts in the space $\mathcal{Z} = \mathcal{X}^N \times \mathcal{X}^N$, then from the proof of Theorem 3.2, we know that T^k is η_k -averaged, where $\eta_k = (2 - \bar{\eta}_k)^{-1}$ and $\bar{\eta}_k = \frac{L}{2}(\tau_k^{-1} - \mu_k^{-1})$. Defining the selection operator \mathcal{S}_n on \mathcal{Z} as $\mathcal{S}_n(y, x) = (y_n, x_n)$, we obtain that $\mathcal{Z} = \mathcal{S}_1(\mathcal{Z}) \times \dots \times \mathcal{S}_N(\mathcal{Z})$ up to an element reordering. To be compatible with the notations of Section 4.1, we assume that $J = N$ and that the random sequence ζ^k driving the SMPDSDS algorithm is set valued in $\{\{1\}, \dots, \{N\}\} \subset 2^{\mathcal{J}}$. In order to establish Theorem 5.2, we need to show that the iterates (y^{k+1}, x^{k+1}) provided by the SMPDSDS algorithm are those who satisfy the equation $(y^{k+1}, x^{k+1}) = T^{k, (\zeta^{k+1})}(y^k, x^k)$. By the direct application of Theorem 4.1, we can obtain Theorem 5.2.

Let us start with the y -update equation. Since $h = \iota_C$, its Legendre-Fenchel transform is $h^* = \iota_{C^\perp}$ where C^\perp is the orthogonal complement of C in \mathcal{X}^N . Consequently,

If we write $(\zeta^{k+1}, v^{k+1}) = T^k(y^k, x^k)$, then by Eq. (3.24a),

$$\zeta_n^{k+1} = y_n^k - \bar{y}^k + \frac{(x_n^k - \bar{x}^k)}{\mu_k} \quad n = 1, \dots, N.$$

Observe that in general, $\bar{y}^k \neq 0$ because in the SMPDSDS algorithm, only one component is updated at a time. If $\{n\} = \zeta^{k+1}$, then $y_n^{k+1} = \zeta_n^{k+1}$ which is Eq. (5.5a). All other components of y^k are carried over to y^{k+1} .

By Equation (3.24b) we also get

$$v_n^{k+1} = \text{prox}_{\tau_k g_n}[x_n^k - \tau_k \nabla f_n(x_n^k) - \tau_k(2y_n^{k+1} - y^k)].$$

If $\{n\} = \zeta^{k+1}$, then $x_n^{k+1} = v_n^{k+1}$ can easily be shown to be given by (5.5b). □

6 Distributed optimization

Consider a set of $N > 1$ computing agents that cooperate to solve the minimization Problem (5.1). Here, f_n, g_n are two private functions available at Agent n . Our purpose is to introduce a random distributed algorithm to solve (5.1). The algorithm is asynchronous in the sense that some components of the network are allowed to wake up at random and perform local updates, while the rest of the network stands still. No coordinator or global clock is needed. The frequency of activation of the various network components is likely to vary.

The examples of this problem appear in learning applications where massive training data sets are distributed over a network and processed by distinct machines [13], [14], in resource allocation problems for communication networks [15], or in statistical estimation problems by sensor networks [16], [17].

6.1 Network model and problem formulation

We consider the network as a graph $G = (Q, E)$ where $Q = \{1, \dots, N\}$ is the set of agents/nodes and $E \subset \{1, \dots, N\}^2$ is the set of undirected edges. We write $n \sim m$

whenever $n, m \in E$. Practically, $n \sim m$ means that agents n and m can communicate with each other.

Assumption 6.1. *G is connected and has no self loop.*

Now we introduce some notations. For any $x \in \mathcal{X}^{|Q|}$, we denote by x_n the components of x , i.e., $x = (x_n)_{n \in Q}$. We regard the functions f and g on $\mathcal{X}^{|Q|} \rightarrow (-\infty, +\infty]$ as $f(x) = \sum_{n \in Q} f_n(x_n)$ and $g(x) = \sum_{n \in Q} g_n(x_n)$. So the Problem (5.1) is equal to the minimization of $f(x) + g(x)$ under the constraint that all components of x are equal.

Next we write the latter constraint in a way that involves the graph G . We replace the global consensus constraint by a modified version of the function ι_C . The purpose of us is to ensure global consensus through local consensus over every edge of the graph.

For any $\epsilon \in E$, say $\epsilon = \{n, m\} \in Q$, we define the linear operator $D_\epsilon(x) : \mathcal{X}^{|Q|} \rightarrow \mathcal{X}^2$ as $D_\epsilon(x) = (x_n, x_m)$ where we assume some ordering on the nodes to avoid any ambiguity on the definition of D . We construct the linear operator $D : \mathcal{X}^{|Q|} \rightarrow \mathcal{Y} \triangleq \mathcal{X}^{2|Q|}$ as $D(x) = (D_\epsilon(x))_{\epsilon \in E}$ where we also assume some ordering on the edges. Any vector $y \in \mathcal{Y}$ will be written as $y = (y_\epsilon)_{\epsilon \in E}$ where, writing $\epsilon = \{n, m\} \in E$, the component y_ϵ will be represented by the couple $y_\epsilon = (y_\epsilon(n), y_\epsilon(m))$ with $n < m$. We also introduce the subspace of \mathcal{X}^2 defined as $\mathcal{C}_2 = \{(x, x) : x \in \mathcal{X}\}$. Finally, we define $h : \mathcal{Y} \rightarrow (-\infty, +\infty]$ as

$$h(y) = \sum_{\epsilon \in E} \iota_{\mathcal{C}_2}(y_\epsilon). \quad (6.1)$$

Then we consider the following problem:

$$\min_{x \in \mathcal{X}^{|Q|}} f(x) + g(x) + (h \circ D)(x). \quad (6.2)$$

Lemma 6.1. ([2]). *Let Assumptions 6.1 hold true. The minimizers of (6.2) are the tuples (x^*, \dots, x^*) where x^* is any minimizer of (5.1).*

6.2 Instantiating the ADMMDS⁺

Now we use the ADMMDS⁺ to solve the Problem (6.2). Since the newly defined function h is separable with respect to the $(y_\epsilon)_{\epsilon \in E}$, we get

$$prox_{\tau_k h}(y) = (prox_{\tau_k \iota_{C_2}}(y_\epsilon))_{\epsilon \in E} = ((\bar{y}_\epsilon, \bar{y}_\epsilon))_{\epsilon \in E}$$

where $\bar{y}_\epsilon = (y_\epsilon(n) + y_\epsilon(m))/2$ if $\epsilon = \{n, m\}$. With this at hand, the update equation (a) of the ADMMDS⁺ can be written as

$$z^{k+1} = ((\bar{z}_\epsilon^{k+1}, \bar{z}_\epsilon^{k+1}))_{\epsilon \in E},$$

where

$$\bar{z}^{k+1} = \frac{x_n^k + x_m^k}{2} + \frac{\mu_k(y_\epsilon^k(n) + y_\epsilon^k(m))}{2}$$

for any $\epsilon = \{n, m\} \in E$. Plugging this equality into Eq. (b) of the ADMMDS⁺, it can be seen that $y_\epsilon^k(n) = -y_\epsilon^k(m)$. Therefore

$$\bar{z}^{k+1} = \frac{x_n^k + x_m^k}{2},$$

for any $k \geq 1$. Moreover

$$y_\epsilon^{k+1} = \frac{x_n^k - x_m^k}{2\mu_k} + y_\epsilon^k(n).$$

Observe that the n -th component of the vector $D^T D x$ coincides with $d_n x_n$, where d_n is the degree (i.e., the number of neighbors) of node n . From (d) of the ADMMDS⁺, the n^{th} component of x^{k+1} can be written

$$x_n^{k+1} = prox_{\tau_k g_n/d_n} \left[\frac{(D^*(u^{k+1} - \tau_k y^{k+1}))_n - \tau_k \nabla f_n(x_n^k)}{d_n} \right],$$

where for any $y \in \mathcal{Y}$,

$$(D^T y)_n = \sum_{m: \{n, m\} \in E} y_{\{n, m\}}(n)$$

is the n -th component of $D^T y \in \mathcal{X}^{|Q|}$. Plugging Eq. (c) of the ADMMDS⁺ together with the expressions of $\bar{z}_{\{n, m\}}^{k+1}$ and $y_{\{n, m\}}^{k+1}$ in the argument of $prox_{\tau_k g_n/d_n}$, we can have

$$x_n^{k+1} = prox_{\tau_k g_n/d_n} \left[(1 - \tau_k \mu_k^{-1}) x_n^k - \frac{\tau_k}{d_n} \nabla f_n(x_n^k) + \frac{\tau_k}{d_n} \sum_{m: \{n, m\} \in E} (\mu_k^{-1} x_m^k - y_{\{n, m\}}^k(n)) \right].$$

The algorithm is finally described by the following procedure: Prior to the clock tick $k + 1$, the node n has in its memory the variables x_n^k , $\{y_{\{n, m\}}^k(n)\}_{m \sim n}$, and $\{x_m^k\}_{m \sim n}$.

Algorithm 6 Distributed ADMMDS⁺.

Initialization: Choose $x^0 \in \mathcal{X}$, $y^0 \in \mathcal{Y}$, s.t. $\sum_n y_n^0 = 0$.

Do

- For any $n \in Q$, Agent n performs the following operations :

$$y_{\{n,m\}}^{k+1}(n) = y_{\{n,m\}}^k(n) + \frac{x_n^k - x_m^k}{2}, \quad \text{for all } m \sim n, \quad (6.3a)$$

$$x_n^{k+1} = \text{prox}_{\tau_k g_n / d_n} \left[(1 - \tau_k \mu_k^{-1}) x_n^k - \frac{\tau_k}{d_n} \nabla f_n(x_n^k) + \frac{\tau_k}{d_n} \sum_{m: \{n,m\} \in E} (\mu_k^{-1} x_m^k - y_{\{n,m\}}^k(n)) \right]. \quad (6.3b)$$

- Agent n sends the parameter y_n^{k+1}, x_n^{k+1} to their neighbors respectively.
 - Increment k .
-

Theorem 6.1. Assume that the minimization Problem (5.1) is consistent, $\liminf_{k \rightarrow \infty} \mu_k > 0$, and $\liminf_{k \rightarrow \infty} \tau_k > 0$. Let Assumption 5.1 and Assumption 6.1 hold true and $\frac{1}{\liminf_{k \rightarrow \infty} \tau_k} - \frac{1}{\liminf_{k \rightarrow \infty} \mu_k} > \frac{L}{2}$. Let $(x^k)_{k \in \mathbb{N}}$ be the sequence generated by Distributed ADMMDS⁺ for any initial point (x^0, y^0) . Then for all $n \in Q$ the sequence $(x_n^k)_{k \in \mathbb{N}}$ converges to a solution of Problem (5.1).

6.3 A Distributed asynchronous primal-dual splitting algorithm with dynamic stepsize

In this section, we use the randomized coordinate descent on the above algorithm, we call this algorithm as distributed asynchronous primal-dual splitting algorithm with dynamic stepsize (DASPDSDS). This algorithm has the following attractive property: Firstly, at each iteration, a single agent, or possibly a subset of agents chosen at random, are activated. Moreover, in the algorithm the coefficient τ , σ is made iteration-dependent to solve the general Problem (5.1), errors are allowed in the evaluation of the operators $\text{prox}_{\sigma h^*}$, $\text{prox}_{\tau g_n}$ and ∇f_n . The errors allow for some tolerance in the numerical implementation of the algorithm, while the flexibility introduced by the iteration-dependent parameters τ_k and σ_k can be used to improve its convergence pattern. Finally, if we let $(\zeta^k)_{k \in \mathbb{N}}$ be a sequence of i.i.d. random variables valued in 2^Q . The value taken by ζ^k represents the agents that will be activated and perform a prox on their x variable at moment k . The asynchronous algorithm goes as follows:

Algorithm 7 DASPDSDS.

Initialization: Choose $x^0 \in \mathcal{X}$, $y^0 \in \mathcal{Y}$.

Do

- Select a random set of agents $\zeta^{k+1} = \mathcal{B}$.
 - For any $n \in \mathcal{B}$, Agent n performs the following operations :
 - For all $m \sim n$, do

$$y_{\{n,m\}}^{k+1}(n) = \frac{y_{\{n,m\}}^k(n) - y_{\{n,m\}}^k(m)}{2} + \frac{x_n^k - x_m^k}{2},$$
 - $x_n^{k+1} = \text{prox}_{\tau_k g_n / d_n} [(1 - \tau_k \mu_k^{-1})x_n^k - \frac{\tau_k}{d_n} \nabla f_n(x_n^k) + \frac{\tau_k}{d_n} \sum_{m: \{n \sim m\} \in E} (\mu_k^{-1} x_m^k + y_{\{n,m\}}^k(m))].$
 - For all $m \sim n$, send $\{x_n^{k+1}, y_{\{n,m\}}^{k+1}(n)\}$ to Neighbor m .
 - For any agent $n \notin \mathcal{B}$, $x_n^{k+1} = x_n^k$, and $y_{\{n,m\}}^{k+1}(n) = y_{\{n,m\}}^k(n)$ for all $m \sim n$.
 - Increment k .
-

Assumption 6.2. The collections of sets $\{\mathcal{B}_1, \mathcal{B}_2, \dots\}$ such that $\mathbb{P}[\zeta^1 = \mathcal{B}_i]$ is positive satisfies $\bigcup \mathcal{B}_i = Q$.

Theorem 6.2. Assume that the minimization Problem (5.1) is consistent, $\liminf_{k \rightarrow \infty} \mu_k > 0$, and $\liminf_{k \rightarrow \infty} \tau_k > 0$. Let Assumption 5.1, Assumption 6.1 and 6.2 hold true, and $\frac{1}{\liminf_{k \rightarrow \infty} \tau_k} - \frac{1}{\liminf_{k \rightarrow \infty} \mu_k} > \frac{L}{2}$. Let $(x_n^k)_{n \in Q}$ be the sequence generated by DASPDSDS for any initial point (x^0, y^0) . Then the sequence $x_1^k, \dots, x_{|Q|}^k$ converges to a solution of Problem (5.1).

Proof. Let $(\bar{f}, \bar{g}, h) = (f \circ D^{-1}, g \circ D^{-1}, h)$ where f, g, h and D are the ones defined in the Problem 6.2. By Equations (3.24a). We write these equations more compactly as $(y^{k+1}, x^{k+1}) = T^k(y^k, x^k)$, the operator T^k acts in the space $\mathcal{Z} = \mathcal{Y} \times \mathcal{R}$, and \mathcal{R} is the image of $\mathcal{X}^{|Q|}$ by D . then from the proof of Theorem 3.2, we know T^k is η_k -averaged operator. Defining the selection operator \mathcal{S}_n on \mathcal{Z} as $\mathcal{S}_n(y, Dx) = (y_\epsilon(n)_{\epsilon \in Q: n \in \epsilon}, x_n)$. So, we obtain that $\mathcal{Z} = \mathcal{S}_1(\mathcal{Z}) \times \dots \times \mathcal{S}_{|Q|}(\mathcal{Z})$ up to an element reordering. Identifying the set \mathcal{J} introduced in the notations of Section 4.1 with Q , the operator $T^{(\zeta^k)}$ is defined as follows:

$$\mathcal{S}_n(T^{(\zeta^k)}(y, Dx)) = \begin{cases} \mathcal{S}_n(T^k(y, Dx)), & \text{if } n \in \zeta^k, \\ \mathcal{S}_n(y, Dx), & \text{if } n \notin \zeta^k. \end{cases}$$

Then by Theorem 4.1, we know the sequence $(y^{k+1}, Dx^{k+1}) = T^{k,(\zeta^{k+1})}(y^k, Dx^k)$ converges almost surely to a solution of Problem (3.25). Moreover, from Lemma 6.1, we have the sequence x^k converges almost surely to a solution of Problem (5.1). Therefore we need to show that the operator $T^{k,(\zeta^{k+1})}$ is translated into the DASPDSDS algorithm. The definition (6.1) of h shows that

$$h^*(\varphi) = \Sigma_{\epsilon \in E} \iota_{\mathcal{C}_2^\perp}(\varphi_\epsilon),$$

where $\mathcal{C}_2^\perp = \{(x, -x) : x \in \mathcal{X}\}$. Therefore, writing

$$(\zeta^{k+1}, v^{k+1} = Dq^{k+1}) = T^k(y^k, \lambda^k = Dx^k),$$

then by Eq. (3.24a),

$$\zeta_\epsilon^{k+1} = \text{proj}_{\mathcal{C}_2^\perp}(y_\epsilon^k + \mu_k^{-1} \lambda_\epsilon^k).$$

Observe that contrary to the case of the synchronous algorithm (6.3), there is no reason here for which $\text{proj}_{\mathcal{C}_2^\perp}(y_\epsilon^k) = 0$. Getting back to $(y^{k+1}, Dx^{k+1}) = T^{k,(\zeta^{k+1})}(y^k, \lambda^k = Dx^k)$, we have for all $n \in \zeta^{k+1}$ and all $m \sim n$,

$$\begin{aligned} y_{\{n,m\}}^{k+1}(n) &= \frac{y_{\{n,m\}}^{k+1}(n) - y_{\{n,m\}}^{k+1}(m)}{2} + \frac{\lambda_{\{n,m\}}^{k+1}(n) - \lambda_{\{n,m\}}^{k+1}(m)}{2} \\ &= \frac{y_{\{n,m\}}^{k+1}(n) - y_{\{n,m\}}^{k+1}(m)}{2} + \frac{x_n^k - x_m^k}{2}. \end{aligned}$$

By Equation (3.24b) we also get

$$v^{k+1} = \arg \min_{w \in \mathcal{R}} [\bar{g}(w) + \langle \nabla \bar{f}(y^k), w \rangle + \frac{\|w - \lambda^k + \tau_k(2y^{k+1} - y^k)\|^2}{2\tau_k}].$$

Upon noting that $\bar{g}(Dx) = g(x)$ and $\langle \nabla \bar{f}(\lambda^k), Dx \rangle = \langle (D^{-1})^* \nabla f(D^{-1}Dx^k), Dx \rangle = \langle \nabla f(x^k), x \rangle$, the above equation becomes

$$q_n^{k+1} = \arg \min_{w \in \mathcal{X}} [g(w) + \langle \nabla f(x^k), w \rangle + \frac{\|D(w - x^k) + \tau_k(2y^{k+1} - y^k)\|^2}{2\tau_k}].$$

Recall that $(D^*Dx)_n = d_n x_n$. Hence, for all $n \in \zeta^{k+1}$, we get after some computations

$$x_n^{k+1} = \text{prox}_{\tau_k g_n / d_n} [x_n^k - \frac{\tau_k}{d_n} \nabla f_n(x_n^k) + \frac{\tau_k}{d_n} (D^*(2y^{k+1} - y^k))_n].$$

Using the identity $(D^*y)_n = \sum_{m: \{n,m\} \in E} y_{\{n,m\}}(n)$, it can easy check these equations coincides with the x -update in the DASPDSDS algorithm. □

7 Numerical experiments

In this section, we present some numerical experiments to verify the effective of our proposed iterative algorithms. All experiments were performed in MATLAB (R2013a) on Lenovo laptop with Intel (R) Core(TM) i7-4712MQ 2.3GHz and 4GB memory on the windows 7 professional operating system.

We consider the following l_1 -regularization problem,

$$\min_{x \in R^n} \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1, \quad (7.1)$$

where $\lambda > 0$ is the regularization parameter, the system matrix $A \in R^{m \times n}$, $b \in R^m$ and $x \in R^n$. Let $\{W_i\}_{i=1}^N$ be a partition of $\{1, 2, \dots, m\}$, the optimization problem (7.1) then writes,

$$\min_{x \in R^n} \sum_{k=1}^N \sum_{i \in W_k} \frac{1}{2} \|A_i x - b_i\|_2^2 + \lambda \|x\|_1. \quad (7.2)$$

Further, splitting the problem (7.2) between the batches, we have

$$\min_{x \in R^{Nn}} \sum_{k=1}^N \left(\sum_{i \in W_k} \frac{1}{2} \|A_i x - b_i\|_2^2 + \frac{\lambda}{N} \|x\|_1 \right) + \iota_C(x), \quad (7.3)$$

where $x = (x_1, x_2, \dots, x_N)$ is in R^{Nn} .

We first describe how the system matrix A and a K -sparse signals x were generated. Let the sample size $m = 1/4n$ and $K = 1/64n$. The system matrix A is random generated from Gaussian distribution with 0 mean and 1 variance. The K -sparse signal x is generated by random perturbation with K values nonzero which are obtained with uniform distribution in $[-2, 2]$ and the rest are kept with zero. Consequently, the observation vector $b = Ax + \delta$, where δ is added Gaussian noise with 0 mean and 0.05 standard variance. Our goal is to recover the sparse signal x from the observation vectors b .

To measure the performance of the proposed algorithms, we use ℓ_2 -norm error between the reconstructed variable x_{rec} and the true variable x_{true} , function values ($fval$) and iteration numbers (k). That is,

$$Err = \|x_{rec} - x_{true}\|_2, \quad fval = \frac{1}{2} \|Ax_{rec} - b\|_2^2.$$

We set the stopping criteria as

$$\frac{\|x^{k+1} - x^k\|_2}{\|x^k\|_2} < \epsilon,$$

where ϵ is a given small constant; Otherwise, the maximum iteration numbers 40000 reached.

Table 1: Numerical results obtained by Algorithm 4

Problem size	Block size	$\epsilon = 10^{-5}$			$\epsilon = 10^{-6}$			$\epsilon = 10^{-8}$		
		<i>Err</i>	<i>fval</i>	<i>k</i>	<i>Err</i>	<i>fval</i>	<i>k</i>	<i>Err</i>	<i>fval</i>	<i>k</i>
$n = 10240$	$N = 2$	0.0616	0.2970	20343	0.0488	0.2901	20839	0.0479	0.2890	22777
	$N = 4$	0.0949	0.2869	40285	0.0497	0.2911	41364	0.0480	0.2890	44771
$n = 20480$	$N = 4$	0.8746	0.2710	68008	0.0511	0.3063	73661	0.0477	0.3046	78879
	$N = 6$									

Table 2: Numerical results obtained by Algorithm 5

Problem size	Block size	$\epsilon = 10^{-5}$			$\epsilon = 10^{-6}$			$\epsilon = 10^{-8}$		
		<i>Err</i>	<i>fval</i>	<i>k</i>	<i>Err</i>	<i>fval</i>	<i>k</i>	<i>Err</i>	<i>fval</i>	<i>k</i>
$n = 10240$	$N = 2$	0.0469	0.4599	—	0.0469	0.3101	—	0.0475	0.2596	—
	$N = 4$	0.0460	0.5515	—	0.0465	0.3479	—	0.0465	0.4180	—

Acknowledgements

This work was supported by the National Natural Science Foundation of China (11131006, 41390450, 91330204, 11401293), the National Basic Research Program of China (2013CB 329404), the Natural Science Foundations of Jiangxi Province (CA201107114, 20114BAB 201004).

References

- [1] L. Condat, A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms, *Journal of Optimization Theory and Applications*, vol. 158, no. 2, pp. 460C479, 2013.

- [2] Bianchi P, Hachem W and Iutzeler F 2014 A Stochastic coordinate descent primal-dual algorithm and applications to large-scale composite (arXiv:1407.0898v1 [math.OC] 3 Jul 2014) Optimization
- [3] Bauschke, H.H., Combettes, P.L.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer, New York (2011)
- [4] Ogura, N., Yamada, I.: Non-strictly convex minimization over the fixed point set of an asymptotically shrinking nonexpansive mapping. Numer. Funct. Anal. Optim. 23(1C2), 113-137 (2002)
- [5] Byrne, C.: A unified treatment of some iterative algorithms in signal processing and image reconstruction. Inverse Probl. 20, 103-120 (2004)
- [6] Combettes, P.L.: Solving monotone inclusions via compositions of nonexpansive averaged operators. Optimization 53, 475-504 (2004)
- [7] Geobel, K., Kirk, W.A.: Topics in Metric Fixed Point Theory. Cambridge Studies in Advanced Mathematics, vol. 28. Cambridge University Press, Cambridge (1990)
- [8] Bruck R E and Passty G B 1979 Almost convergence of the infinite product of resolvents in Banach spaces Nonlinear Anal. 3 279-282.
- [9] Bruck R E and Reich S 1977 Nonexpansive projections and resolvents in Banach spaces Houston J. Math. 3 459-470.
- [10] R. T. Rockafellar, Convex analysis, Princeton Mathematical Series, No. 28. Princeton University Press, Princeton, N.J., 1970.
- [11] I. Daubechies, M. Defrise, and C. De Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, Communications on pure and applied mathematics, vol. 57, no. 11, pp. 1413-1457, 2004.
- [12] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society. Series B (Methodological), pp. 267-288, 1996.

- [13] Forero, P A, Cano A and Giannakis G B 2010 Consensus-based distributed support vector machines The Journal of Machine Learning Research 99 1663-1707.
- [14] Agarwal A, Chapelle O, Dudík M, and Langford J 2011 A reliable effective terascale linear learning system arXiv preprint arXiv:1110.4198.
- [15] P. Bianchi and J. Jakubowicz, Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization, IEEE Transactions on Automatic Control, vol. 58, no. 2, pp. 391- 405, February 2013.
- [16] S.S. Ram, V.V. Veeravalli, and A. Nedic, Distributed and recursive parameter estimation in parametrized linear state-space models, IEEE Trans. on Automatic Control, vol. 55, no. 2, pp. 488-492, 2010.
- [17] P. Bianchi, G. Fort, and W. Hachem, Performance of a distributed
- [18] Yu. Nesterov, Efficiency of coordinate descent methods on huge-scale optimization problems, SIAM Journal on Optimization, vol. 22, no. 2, pp. 341-362, 2012.
- [19] O. Fercoq and P. Richtarik, Accelerated, parallel and proximal coordinate descent, arXiv preprint arXiv:1312.5799, 2013.
- [20] M. Bacak, The proximal point algorithm in metric spaces, Israel Journal of Mathematics, vol. 194, no. 2, pp. 689-701, 2013.